

Pengenalan Karakter Teks Menggunakan Metode *Neural Network Backpropagation*

Titis Hayuning Widya Pramesti,
email: titishayuning@gmail.com
Jurusan Teknik Elektro, Fakultas Teknik Universitas Brawijaya

Abstrak—Pengenalan karakter (*Optical Character Recognition*) merupakan salah satu cabang dari pengenalan pola. Salah satu metode pengenalan pola yang saat ini banyak digunakan adalah *Neural Network Backpropagation* yang sistem kerjanya seperti otak manusia yang dapat mengenali pola setelah mendapat training atau pelatihan terhadap beberapa contoh yang diberikan terlebih dahulu.

Tugas akhir ini lebih berfokus pada pengenalan pola pada karakter. Untuk mendapatkan pola pada setiap karakter diperlukan beberapa fitur dari karakter itu sendiri. Fitur yang diambil pada sebuah karakter meliputi jumlah segmen pembentuk karakter dan bentuk segmennya yang secara garis besar dibedakan menjadi garis, kurva, atau loop. Pengambilan fitur menggunakan metode fuzzy untuk dapat membedakan bentuk-bentuk segmennya sesuai dengan parameter yang diberikan. Sistem pengenalan karakter ini memiliki prosentase keberhasilan dalam pengenalan karakter sekitar 80.12% dengan kemampuannya mengenali 647 data dari 800 data yang terdiri atas data training dan data uji.

Kata Kunci—*Backpropagation, Neural Network, OCR, pengenalan pola.*

I. PENDAHULUAN

OCR atau *Optical Character Recognition* termasuk pada pengenalan pola (*pattern recognition*) di mana sebuah sistem pengenalan pola akan mencoba membaca / mengenali apakah citra masukan yang diterima cocok dengan salah satu citra yang telah ditentukan. Pengenalan pola banyak diaplikasikan untuk mendeteksi sidik jari, tulisan, tanda tangan, bahkan wajah seseorang. Namun, sistem aplikasi OCR yang akan dibangun ini adalah khusus untuk mengenali tulisan teks cetak yang diketik dan dicetak menggunakan printer. Dokumen fisik tersebut adakalanya dikehendaki menjadi dokumen digital, saat softcopy dari data tersebut hilang atau terhapus, OCR dapat dengan mudah membantu mengatasi problem tersebut.

Masalah utama yang diangkat dalam tugas akhir ini adalah bagaimana cara sistem dapat mengenali sebuah citra karakter untuk kemudian diubah menjadi teks ketik

yang standar. Tugas akhir ini lebih berorientasi pada pengenalan fiturnya, sehingga pembahasan lebih ditekankan pada ekstraksi fitur bentuk geometri.

II. DASAR TEORI

2.1 Pengertian Pengenalan Karakter

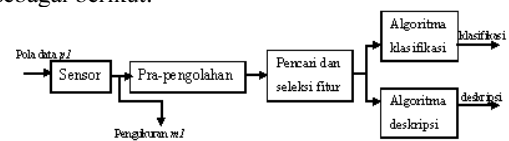
Pengenalan karakter yang umum disebut sebagai OCR (*Optical Character Recognition*) termasuk pada pengenalan pola, yang mana sistem ini khusus digunakan untuk mengenali tulisan teks cetak yang diketik dan dicetak menggunakan printer.

Pengertian pengenalan pola sendiri merupakan suatu sistem yang mencoba untuk membaca / mengenali apakah citra masukan yang diterima cocok dengan salah satu citra yang telah ditentukan. Pengaplikasian pengenalan pola diantaranya seperti pendeteksi sidik jari, tulisan, tanda tangan, bahkan wajah seseorang.

Tujuan dari pengenalan karakter adalah untuk mendukung perkembangan teknologi yang semakin pesat dalam bentuk digital. Sehingga bila terdapat suatu data fisik yang dikehendaki menjadi bentuk digital, maka sistem pengenalan karakter ini dapat dimanfaatkan.

2.2 Struktur Sistem Pengenalan Pola

Secara umum, struktur sistem pengenalan pola adalah sebagai berikut:



Gambar 3.1 Struktur Sistem Pengenalan Pola

a. *Sensor* berfungsi untuk menangkap objek dari dunia nyata dan selanjutnya diubah menjadi sinyal digital melalui proses digitalisasi.

b. *Pra-pengolahan* berfungsi mempersiapkan citra atau sinyal agar dapat menghasilkan ciri yang lebih baik pada tahap berikutnya. Pada tahap ini sinyal informasi ditonjolkan dan sinyal pengganggu (derau) diminimalisasi.

c. *Pencari dan seleksi* fitur berfungsi menemukan karakteristik pembeda yang mewakili sifat utama sinyal sekaligus mengurangi dimensi sinyal menjadi sekumpulan bilangan yang lebih sedikit tetapi representatif.

d. *Algoritma klarifikasi* berfungsi untuk mengelompokkan fitur ke dalam kelas yang sesuai.

berfungsi
sinyal.



e. *Algoritma deskripsi*
memberikan deskripsi pada

2.3 Pendekatan Sistem Pengenalan Pola

Terdapat 3 macam

pendekatan pengenalan pola, yaitu:

a. Pendekatan pengenalan pola statistik (StatPR)

Pengenalan pola statistik memiliki asumsi suatu basis statistik untuk algoritma klasifikasi, sehingga pola dipilah berdasarkan model statistik dari fitur.

b. Pendekatan pengenalan pola sintatik (SyntPR)

Pada pendekatan ini, pola dipilah berdasarkan keserupaan ukuran struktural.

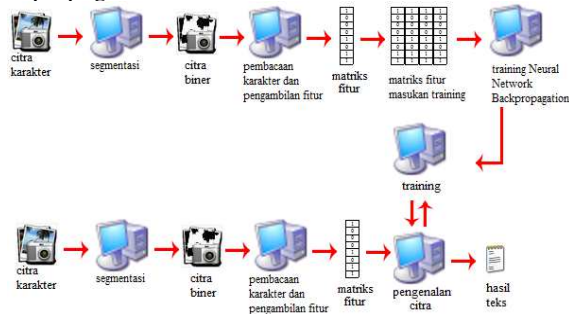
c. Pendekatan pengenalan pola neural (NeuroPR)

Pendekatan ini menggunakan metode jaringan saraf tiruan (JST) untuk mengidentifikasi pola, sehingga pemilahan pola dilakukan berdasarkan tanggapan suatu neuron jaringan pengolah sinyal (neuron) terhadap stimulus masukan (pola).

III. PERANCANGAN DAN IMPLEMENTASI

3.1 Alur Kerja Sistem

Sistem pengenalan pola karakter bekerja berdasarkan serangkaian proses yang ditunjukkan oleh blok diagram pada gambar 3.1. Secara garis besar terdapat 3 proses inti pada sistem ini, yaitu segmentasi, pengambilan fitur dan pengenalan fitur oleh *Neural Network Backpropagation*.

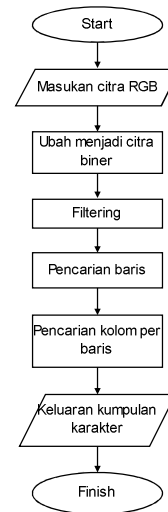


Gambar 3.1 Diagram Blok Sistem

3.2 Segmentasi

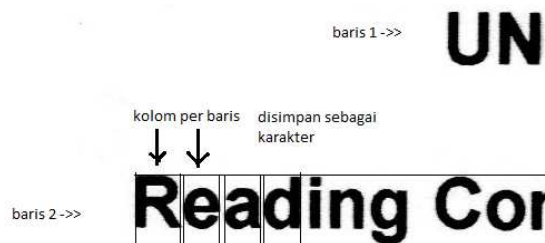
Segmentasi merupakan proses awal pencarian karakter pada dokumen. Proses ini bisa dikatakan sebagai *preprocessing* atau pra-pengolahan. Karena pada tahap ini akan dilakukan normalisasi terlebih dahulu pada dokumen masukan.

Diagram segmentasi terdapat pada Gambar 3.2



Gambar 3.2 Diagram Segmentasi

Proses Segmentasi dimulai dengan memberi masukan sistem berupa citra dokumen asli dengan ekstensi jpg, bmp, tiff, atau png. Citra ini kemudian diubah menjadi citra biner yang hanya memiliki nilai 0 (hitam) dan 1 (putih) saja. Selanjutnya dilakukan proses *filtering* untuk membersihkan *noise*. Dari hasil *filtering* dimulai pembacaan setiap karakter dengan mendeteksi baris pada dokumen dan diikuti pembacaan kolom pada setiap baris yang didapatkan. Proses ini dapat diilustrasikan seperti pada gambar 3.3. Hasil akhir yang didapatkan dari proses segmentasi berupa matrik biner dari setiap karakter yang didapatkan. Contoh salah satu hasil akhir yang didapatkan ditunjukkan pada gambar 3.4.



Gambar 3.3 Proses Segmentasi



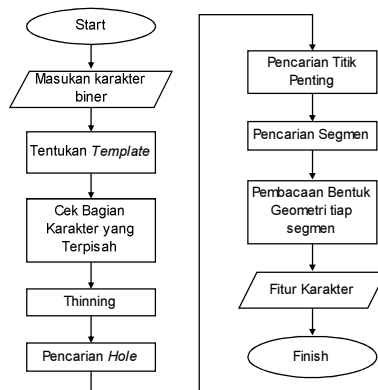
Gambar 3.4 Keluaran segmentasi

3.3 Pengambilan Fitur

Setiap karakter yang didapatkan pada proses segmentasi selanjutnya diambil fiturnya berdasarkan bentuk geometri segmen penyusunnya. Gambar 3.5 menunjukkan diagram alir proses pengambilan fitur.

Namun, terdapat fitur umum selain bentuk geometri segmen yang diambil untuk dapat membantu proses pengenalan. Fitur umum tersebut antara lain adalah

template karakter, adanya bagian karakter yang terpisah, serta keberadaan *hole*.



Gambar 3.5 Diagram alir pengambilan fitur

3.3.1 Fitur Umum

Fitur umum diambil untuk membantu proses pengenalan. Fitur umum ini akan dideteksi ada atau tidaknya pada setiap karakter. Yang termasuk pada fitur ini antara lain *template*, pendeteksian pemisahan bagian karakter, dan *hole*.

Template karakter diambil untuk membedakan huruf besar dan huruf kecil dengan mendeteksi bagian yang kosong pada matrik karakter. Untuk itu *template* diklasifikasikan ke dalam 4 tipe, yaitu kosong bawah, kosong atas, kosong atas bawah, dan terisi penuh.

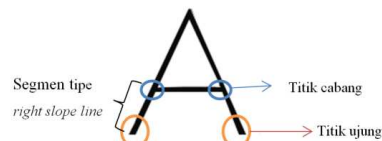
Selanjutnya pengecekan bagian karakter terpisah untuk membantu karakter seperti i, j, ?, !, dll. Bila karakter tersebut terpisah, maka akan dicari pula posisi titiknya.

Sebelum mendapat *hole* maka dilakukan proses *thinning* untuk mendapatkan bentuk *skeleton* citra. Gambar 3.6 menunjukkan proses *thinning*, begitu didapat bentuk *skeleton*, selanjutnya dicek keberadaan *hole* yang mungkin ada dan jumlahnya.



Gambar 3.6 Proses *Thinning*

Sebelum mencari segmen penyusun suatu karakter, perlu dicari terlebih dahulu titik penting karakter. Titik penting yang dimaksud di sini adalah titik ujung dan titik cabang. Kedua titik ini nantinya akan digunakan sebagai titik awal dan titik akhir dalam memulai pencarian tetangga terdekat untuk mengetahui segmen penyusun karakter. Gambar 3.7 menunjukkan contoh bagian yang termasuk titik penting.



Gambar 3.7 Titik penting

Dari setiap titik penting akan ditelusuri segmen-segmen pembentuk karakter. Sebuah segmen akan dimulai dari 1 titik penting ke titik penting lainnya, yang kemudian akan dicari bentuk geometrinya.

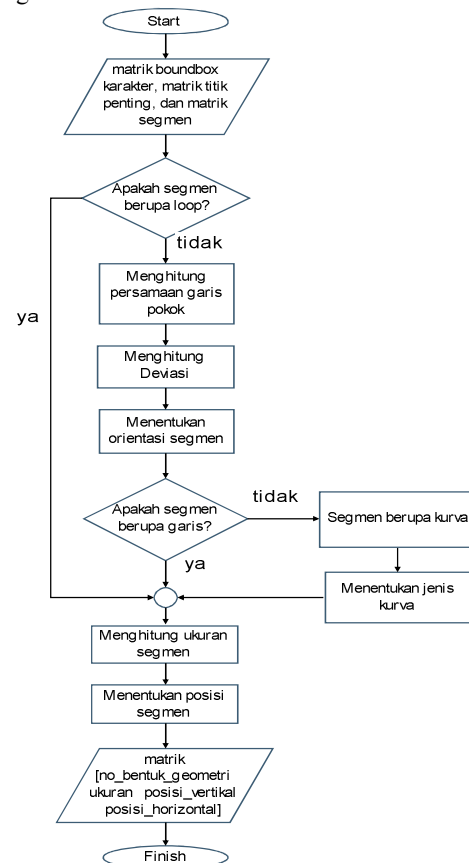
3.3.2 Fitur Bentuk Geometri

Pada pembacaan bentuk geometri, sistem akan mengklasifikasikan sebuah segmen menjadi sebuah bentuk geometri tertentu. Di mana metode yang digunakan adalah metode fuzzy. Bentuk-bentuk geometri yang dimaksud ditunjukkan pada gambar 3.8 di bawah ini.

Klasifikasi Tipe Segmen				
	line	right	left	loop
horizontal	—	⤵	⤴	
vertical		⤶	⤷	○
right slope	↘	⤵	⤴	
left slope	↙	⤵	⤴	

Gambar 3.8 Klasifikasi tipe segmen

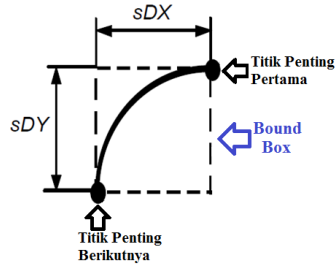
Diagram alir pembacaan bentuk geometri ditunjukkan pada gambar 3.9.



Gambar 3.9 Diagram alir Pembacaan Bentuk Geometri

3.3.2.1 Masukan Pembacaan Bentuk Geometri

Masukan yang diperlukan pada proses ini adalah ukuran bound box dari karakter berupa matrik 2x2, matrik titik penting yang berukuran nx3 (di mana n merupakan jumlah titik penting), dan yang terakhir adalah koordinat-koordinat segmen itu sendiri yang juga berupa matrik berukuran mx2 (m merupakan jumlah elemen segmen). Setiap segmen akan dihitung *bound box*-nya untuk pengklasifikasian ukuran segmen. Gambar 3.10 menunjukkan *bound box* segmen.



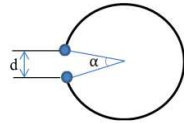
Gambar 3.10 Bound Box Segmen

3.3.2.2 Mengidentifikasi Segmen sebagai Loop

Sebuah segmen perlu dicek terlebih dahulu apakah merupakan sebuah loop atau bukan. Karena untuk sebuah loop hanya perlu melewati tahap pencarian ukuran saja. Untuk mengecek diberlakukan sebuah pertidaksamaan:

$$d \leq \frac{\alpha(k+d)}{360}$$

Dengan d adalah jarak antara titik awal segmen dengan titik akhir segmen. Sedangkan k merupakan jumlah titik pembentuk segmen. Untuk dapat dikatakan sebagai sebuah loop, maka diperlukan sebuah sudut ambang yaitu α . Pada sistem ini, sudut ambang yang digunakan adalah 45° .



Gambar 3.11 Segmen berupa loop

3.3.2.3 Menghitung Persamaan Garis Pokok Segmen

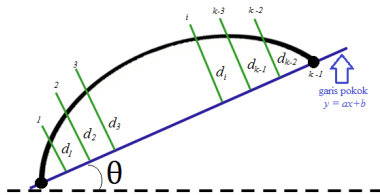
Untuk segmen yang bukan loop, segmen tersebut dapat tergolong garis atau kurva. Untuk dapat menentukan bentuk dasar segmen, perlu dihitung terlebih dahulu persamaan sebuah garis pokok yang menghubungkan antara titik awal dengan titik akhir yang memenuhi persamaan linier:

$$y = ax + b.$$

Setelah itu dihitung jarak rata-rata tiap pixel segmen ke garis pokok (deviasi). Untuk menghitung deviasi berlaku persamaan :

$$D = \frac{\sum_{i=1}^{k-2} d_i}{n}$$

Pada persamaan di atas, D melambangkan deviasi, d merupakan jarak tiap pixel segmen ke garis pokok, dan n adalah banyak pixel yang dihitung. Garis pokok dan deviasi seperti ditunjukkan oleh gambar 3.12



Gambar 3.12 Garis Pokok dan Deviasi

3.3.2.4 Menentukan Bentuk Dasar Segmen

Hasil dari penghitungan deviasi sebelumnya akan digunakan sebagai acuan dalam menentukan bentuk dasar segmen.

Sebuah segmen akan digolongkan ke dalam bentuk garis apabila:

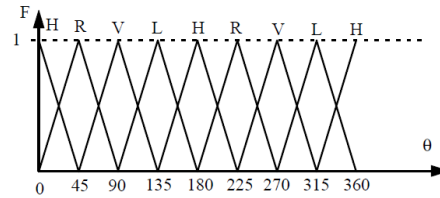
$$D \leq \beta$$

Dengan β merupakan ambang batas kurva. Sedangkan bila hasil perhitungan deviasi menunjukkan angka di atas β maka segmen tersebut akan digolongkan pada bentuk kurva.

3.3.2.5 Menentukan Orientasi Segmen

Orientasi segmen merupakan kemiringan sebuah segmen yang dikelompokkan menjadi 4, yaitu horizontal (H), vertikal (V), serong kiri (L), dan serong kanan (R).

Penentuan orientasi segmen mengikuti peraturan diagram fuzzy pada gambar 3.13 di bawah ini.

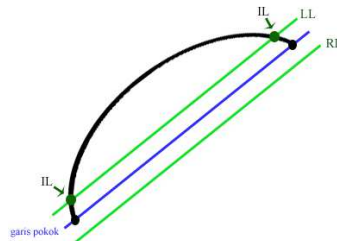


Gambar 3.13 Diagram fungsi fuzzy orientasi segmen

3.3.2.6 Menentukan Jenis Kurva pada Segmen Kurva

Pada segmen berbentuk kurva, perlu dicari jenis dari kurva tersebut. Karena sebuah kurva memiliki sisi terbuka di salah satu sisi, maka jenis kurva hanya dikelompokkan menjadi kurva hadap kanan dan kurva hadap kiri.

Untuk memastikan arah kemiringan kurva, maka diperlukan 2 garis bantu yang sejajar dengan garis pokok. Masing-masing garis bantu tersebut berada di kiri (LL) dan di kanan (RL) garis pokok. Selanjutnya dilihat berapa titik perpotongan (IL / IR) antara garis bantu tersebut dengan kurva. Gambar 3.14 menunjukkan ilustrasi proses ini.



Gambar 3.14 Gambar penentuan jenis kurva

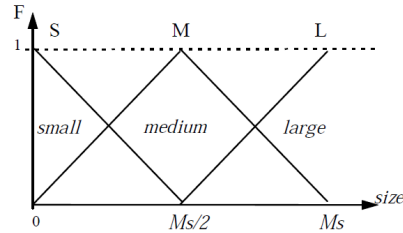
Dari jumlah titik potong garis bantu dengan kurva tersebut, maka jenis kurva dapat ditentukan dengan dua kondisi berikut ini:

- $IL \geq 2$ dan $IR = 0$ maka jenis kurva adalah kurva hadap kanan.
- $IR \geq 2$ dan $IL = 0$ maka jenis kurva adalah kurva hadap kiri.

3.3.2.7 Menentukan Ukuran Segmen

Tahap selanjutnya adalah menentukan ukuran segmen terhadap karakter. Ukuran segmen dikelompokkan menjadi kecil, sedang, dan besar.

Penentuan ukuran akan mengikuti diagram fuzzy pada gambar 3.15.



Gambar 3.15 Diagram fuzzy ukuran segmen

Untuk garis horizontal:

- Size = banyak pixel anggota segmen.
- Ms = lebar karakter.

Untuk garis vertikal:

- Size = banyak pixel anggota segmen.
- Ms = tinggi karakter.

Untuk kurva :

- Size = $sDx \times sDY$, (ukuran *bound box* kurva)
- Ms = $m \times n$, (ukuran karakter)

3.3.2.8 Menentukan Posisi Segmen

Posisi segmen dibagi menjadi dua bagian yaitu secara horizontal (kiri, tengah, kanan) dan secara vertikal (atas, tengah, bawah).

Dalam pencarian posisi akan mengikuti kondisi berikut:

- Kiri/Atas

$$tmin < \frac{1}{3}pk \text{ dan } ps \leq \frac{1}{3}pk$$

- Tengah

$$tmin \leq \frac{1}{3}pk \text{ atau } tmin \leq \frac{2}{3}pk \text{ dan } ps \leq \frac{1}{3}pk$$

- Kanan/Bawah

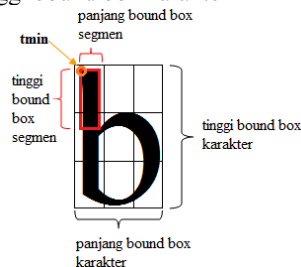
$$tmin > \frac{2}{3}pk \text{ dan } ps > \frac{1}{3}pk$$

Untuk kiri, tengah, dan kanan:

- tmin = posisi titik x minimal segmen
- ps = panjang bound box segmen
- pk = panjang bound box karakter

Untuk atas, tengah, dan bawah:

- tmin = posisi titik y minimal segmen
- ps = tinggi bound box segmen
- pk = tinggi bound box karakter



Gambar 3.16 Penentuan posisi segmen

Hasil dari posisi segmen dapat dilihat pada tabel 3.1.

TABEL 3.1
ILUSTRASI POSISI SEGMENT PADA KARAKTER

Atas kiri	Atas tengah	Atas kanan
Tengah kiri	Tengah tengah	Tengah kanan
Bawah kiri	Bawah tengah	Bawah kanan

3.3.2.9 Keluaran Bentuk Geometri

Keluaran bentuk geometri adalah berupa matrik berukuran 1x4. Di mana kolom pertama menunjukkan angka yang mewakili tiap bentuk geometri, kolom kedua adalah ukuran segmen, kolom ketiga adalah posisi secara vertikal, dan kolom keempat merupakan posisi secara horizontal.

- Contoh:

garis vertikal sedang posisi bawah kiri

[2 2 3 1]

3.4 Keluaran Pembacaan Karakter

Keluaran dari proses pembacaan karakter ini merupakan fitur dari karakter tersebut yang direpresentasikan dalam bentuk matriks berukuran 118x1. Selanjutnya fitur ini akan menjadi masukan bagi jaringan *Neural Network* yang bertugas sebagai pengenalan karakter.

Format fitur keluaran dari karakter telah ditetapkan berdasarkan posisi segmen pada karakter. Nilai pada fitur karakter akan bernilai 1 bila fitur tersebut ada dalam karakter, dan bernilai 0 untuk sebaliknya. Format fitur ini ditunjukkan sesuai tabel 3.2.

TABEL 3.2
FORMAT FITUR KARAKTER

No.	Fitur Karakter
1.	Template 1
2.	Template 2
3.	Template 3
4.	Template 4
5.	1 Hole
6.	2 Hole
7.	Terpisah vertikal
8.	Terpisah horizontal
9.	Posisi titik di atas
10.	Posisi titik di bawah
11-22 Posisi atas kiri	
11.	Bentuk Geometri 1
12.	Bentuk Geometri 2
13.	Bentuk Geometri 3
14.	Bentuk Geometri 4
15.	Bentuk Geometri 5

16.	Bentuk Geometri 6
17.	Bentuk Geometri 7
18.	Bentuk Geometri 8
19.	Bentuk Geometri 9
20.	Bentuk Geometri 10
21.	Bentuk Geometri 11
22.	Bentuk Geometri 12
23-34	Posisi atas tengah
35-46	Posisi atas kanan
47-58	Posisi tengah kiri
59-70	Posisi tengah tengah
71-82	Posisi tengah kanan
83-94	Posisi bawah kiri
95-106	Posisi bawah tengah
107-118	Posisi bawah kanan

3.5 Pengenalan Karakter Oleh Neural Network Backpropagation

Sebelum dapat melakukan proses pengenalan, jaringan NNB perlu mendapatkan training terlebih dahulu. Proses training jaringan NNB dilakukan agar sistem pengenalan karakter dapat mengenali citra karakter dengan tepat.

Pada tahap ini dimasukkan 400 citra yang akan ditraining. Masukan sejumlah 400 citra tersebut dengan rincian setiap kelompok karakter huruf kecil dan huruf besar masing-masing diwakili dengan 150 contoh citra yang diambil dari berbagai sumber. Sedangkan untuk tanda baca dan angka masing-masing diwakili dengan 50 contoh citra.

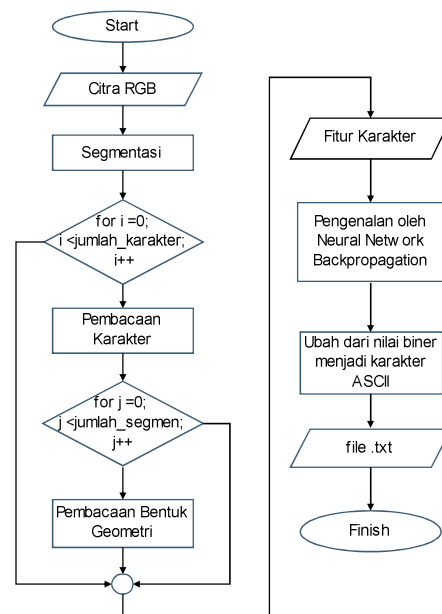
Masukan sistem berupa fitur-fitur dari karakter yang direpresentasikan dengan matrik berukuran 118 x 400. Di mana 118 merupakan fitur biner yang didapatkan dari tiap karakter dan 400 merupakan jumlah total karakter yang akan ditraining.

Target merupakan keluaran yang diharapkan dari hasil training. Karena keluaran yang diharapkan berupa nilai biner dari karakter, untuk itu matrik yang menjadi target training berukuran 7 x 400. Sehingga setiap masukan akan dipasangkan tepat dengan 1 target.

Keluaran sistem berupa sebuah jaringan *neural network backpropagation* yang telah ditraining. Sehingga jaringan ini nantinya dapat digunakan pada sistem pengenalan karakter dengan cara mengenali fitur masukan sistem pengenalan karakter untuk dicocokkan dengan contoh fitur yang telah melalui proses training.

3.6 Perancangan dan Implementasi Sistem Pengenalan Karakter

Perancangan sistem pengenalan karakter adalah program utama yang akan dijalankan untuk mengenali image yang dimasukkan. Diagram alir sistem pengenalan karakter ini ditunjukkan pada gambar 3.17.



Gambar 3.17 Diagram alir Sistem Pengenalan Karakter

3.6.1 Masukan Sistem Pengenalan Karakter

Masukan sistem berupa citra dokumen dengan ekstensi jpg, png, tiff, atau bmp. Gambar akan diubah dalam bentuk matrik citra dengan format RGB dengan ukuran 8 bit.

3.6.2 Segmentasi

Proses awal adalah pencarian karakter-karakter yang terdapat pada dokumen. Dengan dimulai dari pendeteksian baris, kemudian pendeteksian kolom per baris yang mengandung karakter.

3.6.3 Pembacaan Karakter

Setiap karakter yang ditemukan dalam proses segmentasi akan melewati proses pembacaan karakter untuk diambil fiturnya. Pada tugas akhir ini fitur yang diambil adalah bentuk geometri dari tiap segmen pembentuk karakter tersebut. Untuk itu, pada proses pembacaan karakter meliputi proses pembacaan bentuk geometri tiap segmen.

Keluaran dari tahap ini adalah matrik berukuran 118x1 yang telah berisi fitur dari karakter dan menjadi masukan bagi jaringan *Neural Network Backpropagation* untuk dikenali.

3.6.4 Pengenalan oleh Neural Network Backpropagation

Merupakan bagian pengenalan yang dilakukan oleh jaringan *Neural Network Backpropagation* yang sebelumnya telah dilatih dengan beberapa data training.

Masukannya adalah matriks berukuran 118 x 1 yang berisikan fitur-fitur dari karakter. Selanjutnya jaringan akan memutuskan sebuah karakter yang diharapkan merupakan karakter teks dari citra masukan sistem.

Keluaran dari proses ini masih berupa nilai biner dari karakter yang diputuskan sebagai hasil dari pengenalan oleh jaringan NNB yang sudah ditraining ini.

3.6.5 Pengubahan menjadi Karakter ASCII

Dari nilai biner hasil dari pengenalan jaringan *neural*, selanjutnya diubah menjadi teks ASCII yang berupa karakter ketik.

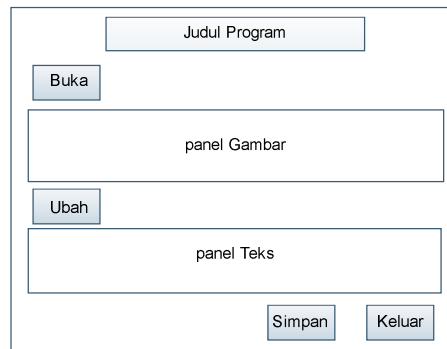
Proses pengubahan adalah dengan mengubah ke dalam bentuk desimal terlebih dahulu, untuk kemudian diubah ke dalam bentuk karakternya.

3.6.6 Keluaran Sistem Pengenalan Karakter

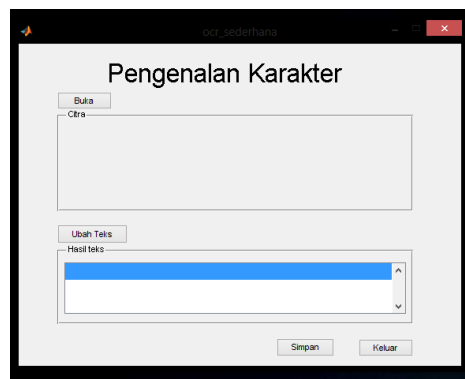
Keluaran dari sistem secara keseluruhan adalah berupa karakter teks ketik yang merupakan hasil pengenalan dari citra masukan dengan format filenya berupa *text*.

3.6.7 User Interface Sistem Pengenalan Karakter

Rancangan *Interface* sistem pengenalan karakter adalah seperti Gambar 3.18. Tampilan memiliki 1 panel citra untuk citra masukan, 1 panel teks untuk hasil keluaran karakter, 1 panel teks untuk keluaran waktu proses, dan 4 tombol untuk membuka citra, mengenali citra, menyimpan hasil pengenalan secara langsung pada bentuk *text*, dan untuk keluar dari program. Hasil dari perancangan dapat dilihat pada Gambar 3.19.



Gambar 3.18 Rancangan *Interface* Sistem Pengenalan Karakter



Gambar 3.19 Implementasi *Interface* Sistem Pengenalan Karakter

IV. PENGUJIAN DAN ANALISIS SISTEM

4.1 Pengujian dan Analisis terhadap Pengenalan Karakter Masukan

Parameter keberhasilan pada sistem adalah besar persen ketepatan pengenalan sistem terhadap beberapa masukan citra karakter. Pengujian dilakukan secara berkelompok. Di mana terdapat 4 kelompok yaitu pengujian pada huruf kecil, huruf besar, angka, dan

tanda baca. Selain itu, pengujian dilakukan pada data training, dan data baru yang belum pernah ditraining.

Contoh pengujian pada sistem yang dilakukan pada setiap kata ditunjukkan seperti pada tabel 4.1 dan tabel 4.2.

TABEL 4.1

TABEL PENGUJIAN DATA TRAINING

No	Citra masukan	Hasil	Benar
1	paragraphs?	poragraphb/?	8
2	English	EngIish	6
3	University	UnivesciIy	7
4	0610630002	061O630002	9

TABEL 4.2

TABEL PENGUJIAN DATA NON TRAINING

No	Citra masukan	Hasil	Benar
1	Example	ExompIe	5
2	quickly	qvicklg	4
3	Engineering	Hngincering	9
4	19710301	1971O30I	6

Pada pengujian setiap kata terlihat bahwa kesalahan sering terjadi pada huruf yang agak mirip. Dengan adanya template sistem masih bisa membedakan antara huruf besar dengan huruf kecil. Tetapi untuk huruf seperti g dan y yang memiliki ekor di bawah sering terjadi kesalahan. Begitu pula antara angka 0, huruf O, maupun huruf o biasanya terjadi kesalahan dikarenakan pembacaan template yang terkadang berbeda. Selain itu antara huruf I, huruf l, dan angka 1 juga masih sering terjadi kesalahan dalam pembacaan. Karena untuk beberapa jenis font, bentuknya juga mirip.

Pada huruf yang agak mirip lainnya seperti b, d, k, h, p, dan q masih terbilang berhasil karena posisi *ascender* (garis vertical atas) maupun *descender* (garis vertical bawah) masih dapat dibedakan.

Untuk tanda baca yang memiliki bentuk amat kecil seperti titik dan koma, masih sering terdapat kesalahan, karena pada proses thinning keduanya dapat memiliki bentuk yang sama, di mana sebuah titik yang tebal adakalanya pada proses thinning menjadi sebuah garis kecil seperti pada tanda baca koma.

Pengujian sistem secara berkelompok adalah untuk menguji keberhasilan sistem secara umum. Hasil dari sistem secara umum ditampilkan pada tabel 4.3.

TABEL 4.3
PROSENTASE RATA-RATA SISTEM

No	Kelompok Pengujian	Data Training yang Dikenali	Data Non Training yang Dikenali	(%)
1	Huruf Kecil (150 data)	125	120	81.6%
2	Huruf Besar (150 data)	122	121	81%
3	Tanda Baca (50 data)	40	38	78%
4	Angka (50 data)	43	38	81%
Prosentase Rata-Rata				80.12%

V. PENUTUP

Berdasarkan hasil perancangan, implementasi, pengujian dan analisis yang dilakukan, maka diambil kesimpulan sebagai berikut:

1. Berdasarkan hasil pengujian sistem terhadap pengenalan citra karakter, citra yang telah ditraining lebih mudah dikenali daripada citra yang tidak melalui proses training.
2. Adanya template membantu dalam memisahkan antara karakter yang mirip maupun antara huruf besar dan huruf kecil, namun ketergantungan pada template juga berdampak kesalahan pembacaan bila satu baris berisi karakter-karakter yang tidak memiliki variasi template.
3. Sistem dapat mengenali dengan lebih baik untuk karakter yang memiliki data training cukup banyak, sedangkan banyaknya data yang dimasukkan dalam training mempengaruhi kinerja jaringan sehingga untuk dapat melatih data yang cukup banyak dan beragam diperlukan *resource* yang lebih besar pula.
4. Sistem memiliki tingkat keberhasilan pengenalan rata-rata sebesar 80.12% dengan kemampuannya mengenali 647 data dari 800 data yang terdiri atas data training dan data uji.

Saran yang dapat diberikan untuk pengembangan sistem pengenalan karakter ini antara lain :

1. Perlu pengambilan fitur yang lebih *general*, mungkin dengan metode atau fitur lain, untuk memperbaiki kinerja sistem, sehingga tidak bergantung pada 1 fitur.
2. Ditambahkan penanganan untuk citra masukan yang berbentuk *italic* atau citra hasil *scanner* yang miring.

VI. DAFTAR PUSTAKA

- [1] J. Gilewski, et al., 1997. "Education Aspects: Handwriting Recognition Neural Networks Fuzzy Logic". Makalah disampaikan pada *IAPR International Conference on Pattern Recognition and Information Processing*. 1997.
- [2] Mori, Sunji dkk. 1999. *Optical Character Recognition*. Canada: John Wiley & Sons.
- [3] Nurwanto, Tri Budi. 2007. "Pengenalan Huruf Tulisan Tangan Menggunakan Logika Fuzzy Dengan Pendekatan Neural

Networks Back Propagation". *Skripsi* tidak diterbitkan. Bandung: Jurusan Teknik Informatika STT Telkom, 2007.

- [4] Putra, Dharma. 2010. *Pengolahan Citra Digital*. Yogyakarta : Penerbit Andi.
- [5] Pramuditya, Yoga. *Sejarah dan Kajian Tipografi*. <http://share.pdfonline.com/a674d46519cc477db51e87c3db86dd80/899EE01.pdf> (diakses 27 Juli 2011)
- [6] Ramza, M.T., Harry. 2007. *Teknik Pemrograman Menggunakan Matlab*. Jakarta: Grasindo.
- [7] Haralambous, Yannis. 2007. *Fonts & Encodings*. United States of America : O'Reilly Media.